Package ‘epiomics’

January 9, 2023

Title Analysis of Omics Data in Observational Studies

Version 0.0.1

Description A collection of fast and flexible functions for analyzing omics data in observational studies. Multiple different approaches for integrating environmental/genetic factors, omics data, and/or phenotype data are implemented. This includes functions for performing omics wide association studies with one or more variables of interest as the exposure or outcome; a function for performing a meet in the middle analysis for linking exposures, omics, and outcomes (as described by Chadeau-Hyam et al., (2010) <doi:10.3109/1354750X.2010.533285>); and a function for performing a mixtures analysis across all omics features using quantile-based g-Computation (as described by Keil et al., (2019) <doi:10.1289/EHP5838>).

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.2.2

LazyData true

Imports data.table, qgcomp, survival, lme4, lmerTest, ggplot2, ggrepel

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author Jesse Goodrich [aut, cre] (<https://orcid.org/0000-0001-6615-0472>)

Maintainer Jesse Goodrich <jagoodri@usc.edu>

Depends R (>= 3.5.0)

Repository CRAN

Date/Publication 2023-01-09 10:30:10 UTC

R topics documented:

`example_data` *Example data with multiple exposures, multiple outcomes,*

Description

Example data with multiple exposures, multiple outcomes,

Usage

```
data(example_data)
```

Format

An dataframe with multiple exposures, outcomes, and omics features.

Examples

```
data(example_data)
```

`meet_in_middle` *Perform 'omics wide association study*

Description

Implements a meet in the middle analysis for identifying omics associated with both exposures and outcomes, as described by Chadeau-Hyam et al., 2010.

Usage

```
meet_in_middle(
  df,
  exposure,
  outcome,
  omics,
  covars = NULL,
  outcome_family = "gaussian",
  confidence_level = 0.95,
  conf_int = FALSE,
  ref_group_exposure = NULL,
  ref_group_outcome = NULL
)
```

Arguments

`df` Dataframe

`exposure` Name of the exposure of interest. Can be either continuous or dichotomous. Currently, only a single exposure is supported.

`outcome` Name of the outcome of interest. Can be either continuous or dichotomous. For dichotomous variables, must set `outcome_family` to "logistic", and values must be either 0/1 or a factor with the first level representing the reference group. Currently, only a single outcome is supported.

`omics` Names of all omics features in the dataset

`covars` Names of covariates (can be NULL)

`outcome_family` "gaussian" for linear models (via lm) or "binomial" for logistic (via glm)

`confidence_level`
Confidence level for marginal significance (defaults to 0.95)

`conf_int` Should Confidence intervals be generated for the estimates? Default is FALSE. Setting to TRUE will take longer. For logistic models, calculates Wald confidence intervals via `confint.default`.

`ref_group_exposure`
Reference category if the exposure is a character or factor. If not, can leave empty.

`ref_group_outcome`
Reference category if the outcome is a character or factor. If not, can leave empty.

Value

A list of three dataframes, containing:

1. Results from the Exposure-Omics Wide Association Study
2. Results from the Omics-Outcome Wide Association Study
3. Overlapping significant features from 1 and 2. For each omics wide association, results are provided in a data frame with 6 columns: feature_name: name of the omics feature estimate: the model estimate for the feature. For linear models, this is the beta: for logistic models, this is the log odds. se: Standard error of the estimate p_value: p-value for the estimate adjusted_pval: FDR adjusted p-value threshold: Marginal significance, based on unadjusted p-values

Examples

```
# Load Example Data
data("example_data")

# Get names of omics
colnames_omic_fts <- colnames(example_data)[grep("feature_",
                                                 colnames(example_data))][1:10]

# Meet in the middle with a dichotomous outcome
```

```
res <- meet_in_middle(df = example_data,
                      exposure = "exposure1",
                      outcome = "disease1",
                      omics = colnames_omic_fts,
                      covars = c("age", "sex"),
                      outcome_family = "binomial")

# Meet in the middle with a continuous outcome
res <- meet_in_middle(df = example_data,
                      exposure = "exposure1",
                      outcome = "weight",
                      omics = colnames_omic_fts,
                      covars = c("age", "sex"),
                      outcome_family = "gaussian")

# Meet in the middle with a continuous outcome and no covariates
res <- meet_in_middle(df = example_data,
                      exposure = "exposure1",
                      outcome = "weight",
                      omics = colnames_omic_fts,
                      outcome_family = "gaussian")
```

owas *Perform 'omics wide association study*

Description

Implements an omics wide association study with the option of using the 'omics data as either the dependent variable (i.e., for performing an exposure –> 'omics analysis) or using the 'omics as the independent variable (i.e., for performing an 'omics –> outcome analysis). Allows for either continuous or dichotomous outcomes, and provides the option to adjust for covariates.

Usage

```
owas(
  df,
  var,
  omics,
  covars = NULL,
  var_exposure_or_outcome,
  family = "gaussian",
  confidence_level = 0.95,
  conf_int = FALSE,
  ref_group = NULL
)
```

Arguments

`df` Dataset

`var` Name of the variable or variables of interest- this is usually either an exposure variable or an outcome variable. Can be either continuous or dichotomous. For dichotomous variables, must set `family` to "binomial", and values must be either 0/1 or a factor with the first level representing the reference group. Can handle multiple variables, but they must all be of the same `family`.

`omics` Names of all omics features in the dataset

`covars` Names of covariates (can be NULL)

`var_exposure_or_outcome`
Is the variable of interest an exposure (independent variable) or outcome (dependent variable)? Must be either "exposure" or "outcome"

`family` "gaussian" (default) for linear models (via lm) or "binomial" for logistic (via glm)

`confidence_level`
Confidence level for marginal significance (defaults to 0.95, or an alpha of 0.05)

`conf_int` Should Confidence intervals be generated for the estimates? Default is FALSE. Setting to TRUE will take longer. For logistic models, calculates Wald confidence intervals via `confint.default`.

`ref_group` Reference category if the variable of interest is a character or factor. If not, can leave empty.

Value

A data frame with 6 columns: feature_name: name of the omics feature estimate: the model estimate for the feature. For linear models, this is the beta; for logistic models, this is the log odds. se: Standard error of the estimate test statistic: t-value p_value: p-value for the estimate adjusted_pval: FDR adjusted p-value threshold: Marginal significance, based on unadjusted p-values

Examples

```
# Load Example Data
data("example_data")

# Get names of omics
colnames_omic_fts <- colnames(example_data)[grep("feature_",
                                              colnames(example_data))][1:10]

# Get names of exposures
expnms = c("exposure1", "exposure2", "exposure3")

# Run function with one continuous exposure as the variable of interest
owas(df = example_data,
     var = "exposure1",
     omics = colnames_omic_fts,
     covars = c("age", "sex"),
     var_exposure_or_outcome = "exposure",
     family = "gaussian")
```

```
# Run function with multiple continuous exposures as the variable of interest
owas(df = example_data,
     var = expnms,
     omics = colnames_omic_fts,
     covars = c("age", "sex"),
     var_exposure_or_outcome = "exposure",
     family = "gaussian")

# Run function with dichotomous outcome as the variable of interest
owas(df = example_data,
     var = "disease1",
     omics = colnames_omic_fts,
     covars = c("age", "sex"),
     var_exposure_or_outcome = "outcome",
     family = "binomial")
```

owas_clogit — *Perform 'omics wide association study for matched case control studies*

Description

Implements an omics wide association study for matched case control studies using conditional logistic regression. For this function, the variable of of interest should be a dichotomous outcome, and the strata is the variable indicating the matching.

Usage

```
owas_clogit(
  df,
  cc_status,
  cc_set,
  omics,
  covars = NULL,
  confidence_level = 0.95,
  conf_int = FALSE,
  method = "efron"
)
```

Arguments

`df` Dataset

`cc_status` Name of the variable indicating case control status. Must be either 0/1 or a factor with the first level representing the reference group.

`cc_set` Name of the variable indicating the case control set.

`omics` Names of all omics features in the dataset reference group.

`covars` Names of covariates (can be NULL)

`confidence_level`
Confidence level for marginal significance (defaults to 0.95, or an alpha of 0.05)

`conf_int` Should Confidence intervals be generated for the estimates? Default is FALSE. Setting to TRUE will take longer. For logistic models, calculates Wald confidence intervals via `confint.default`.

`method` method used the correct (exact) calculation in the conditional likelihood or one of the approximations. Default is "efron". Passed to `clogit`.

Value

A data frame with 6 columns: feature_name: name of the omics feature estimate: the model estimate for the feature. For linear models, this is the beta; for logistic models, this is the log odds. se: Standard error of the estimate test statistic: t-value p_value: p-value for the estimate adjusted_pval: FDR adjusted p-value threshold: Marginal significance, based on unadjusted p-values

`owas_mixed_effects` *Perform ’omics wide association study with linear or generalized mixed models*

Description

Implements an omics wide association study with the option of using the ’omics data as either the dependent variable (i.e., for performing an exposure –> ’omics analysis) or using the ’omics as the independent variable (i.e., for performing an ’omics –> outcome analysis). Allows for either continuous or dichotomous outcomes, and provides the option to adjust for covariates.

Usage

```
owas_mixed_effects(
  df,
  var,
  omics,
  random_effects,
  covars = NULL,
  var_exposure_or_outcome,
  family = "gaussian",
  confidence_level = 0.95,
  conf_int = FALSE,
  REML = TRUE,
  ref_group = NULL
)
```

Arguments

df Dataset

var Name of the variable or variables of interest- this is usually either an exposure variable or an outcome variable. Can be either continuous or dichotomous. For dichotomous variables, must set family to "binomial", and values must be either 0/1 or a factor with the first level representing the reference group. Can handle multiple variables, but they must all be of the same family.

omics Names of all omics features in the dataset

random_effects Random effects, formatted as specified by lmer or glmer

covars Names of covariates (can be NULL)

var_exposure_or_outcome
Is the variable of interest an exposure (independent variable) or outcome (dependent variable)? Must be either "exposure" or "outcome"

family "gaussian" (default) for linear models (via lmer) or "binomial" for logistic (via glmer)

confidence_level
Confidence level for marginal significance (defaults to 0.95, or an alpha of 0.05)

conf_int Should Confidence intervals be generated for the estimates? Default is FALSE. Setting to TRUE will take longer. For logistic models, calculates Wald confidence intervals via confint.default.

REML logical scalar - Should the estimates be chosen to optimize the REML criterion (as opposed to the log-likelihood)? Default is TRUE

ref_group Reference category if the variable of interest is a character or factor. If not, can leave empty.

Value

A data frame with 6 columns: feature_name: name of the omics feature estimate: the model estimate for the feature. For linear models, this is the beta; for logistic models, this is the log odds. se: Standard error of the estimate test statistic: t-value p_value: p-value for the estimate adjusted_pval: FDR adjusted p-value threshold: Marginal significance, based on unadjusted p-values

owas_qgcomp *Perform omics wide association study using qgcomp*

Description

Omics wide association study using quantile-based g-Computation (as described by Keil et al., (2019) doi:10.1289/EHP5838) to examine associations of exposure mixtures with each individual 'omics feature as an outcome 'omics data as either the dependent variable. Allows for either continuous or dichotomous outcomes, and provides the option to adjust for covariates.

Usage

```
owas_qgcomp(df, expnms, omics, covars = NULL, q = 4, confidence_level = 0.95)
```

Arguments

`df` Dataset

`expnms` Name of the exposures. Can be either continuous or dichotomous. For dichotomous variables, must set q to "NULL", and values must be either 0/1.

`omics` Names of all omics features in the dataset

`covars` Names of covariates (can be NULL)

`q` NULL or number of quantiles used to create quantile indicator variables representing the exposure variables. Defaults to 4If NULL, then qgcomp proceeds with un-transformed version of exposures in the input datasets (useful if data are already transformed, or for performing standard g-computation).

`confidence_level`
Confidence level for marginal significance (defaults to 0.95, or an alpha of 0.05)

Value

A data frame with 6 columns: feature_name: name of the omics feature psi: the model estimate for the feature. For linear models, this is the beta; for logistic models, this is the log odds. se: Standard error of the estimate p_value: p-value for the estimate adjusted_pval: FDR adjusted p-value threshold: Marginal significance, based on unadjusted p-values

Examples

```
# Load Example Data
data("example_data")

# Get names of omics
colnames_omic_fts <- colnames(example_data)[grep("feature_",
                                             colnames(example_data))][1:5]

# Names of exposures in mixture
 exposure_names = c("exposure1", "exposure2", "exposure3")

# Run function without covariates
out <- owas_qgcomp(df = example_data,
                   expnms = exposure_names,
                   omics = colnames_omic_fts,
                   q = 4,
                   confidence_level = 0.95)


# Run analysis with covariates
out <- owas_qgcomp(df = example_data,
                   expnms = c("exposure1", "exposure2", "exposure3"),
                   covars = c("weight", "age", "sex"),
                   omics = colnames_omic_fts,
                   q = 4,
                   confidence_level = 0.95)
```

volcano_owas *Create volcano plot using results from owas*

Description

Creates a volcano plot based on ggplot using the results from the owas function.

Usage

```
volcano_owas(
  df,
  annotate_ftrs = TRUE,
  annotation_p_threshold = 0.05,
  highlight_adj_p = TRUE,
  highlight_adj_p_threshold = 0.05,
  horizontal_line_p_value = 0.05
)
```

Arguments

df output from owas function call

annotate_ftrs Should features be annotated with the feature name? Default is TRUE. If necessary can change the p_value_threshold as well.

annotation_p_threshold
If annotate_ftrs = TRUE, can set annotation_p_threshold to change the p-value threshold for which features will be annotated. Defaults to 0.05.

highlight_adj_p
Should features which meet a specific adjusted p-value threshold be highlighted? Default is TRUE.

highlight_adj_p_threshold
If highlight_adj_p = TRUE, can set annotation_adj_p_threshold to change the adjusted p-value threshold for which features will be highlighted. Defaults to 0.05.

horizontal_line_p_value
Set the p-value for the horizontal line for the threshold of significance.

Value

A ggplot figure

Examples

```
data("example_data")

# Get names of omics
colnames_omic_fts <- colnames(example_data)[
  grep("feature_",
       colnames(example_data))][1:5]

# Run function with continuous exposure as the variable of interest
owas_out <- owas(df = example_data,
                 var = "exposure1",
                 omics = colnames_omic_fts,
                 covars = c("age", "sex"),
                 var_exposure_or_outcome = "exposure",
                 family = "gaussian")

vp <- volcano_owas(owas_out)
```

Index